

CLAIM AMENDMENTS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Previously Presented) A method, comprising:
 - loading a virtual machine monitor (VMM) to support a plurality of virtual machines in a computer system, the VMM including a VMM multiplexer;
 - loading a first and a second virtual machine (VM) supported by the VMM;
 - determining a first VM platform configuration including a first hash value based on information measured from the first VM and a second VM platform configuration including a second hash value based on information measured from the second VM;
 - using a trusted hardware device shared between the first and the second VM to compute a compound hash value based on a combination of the first VM platform configuration including the first hash value and the second VM platform configuration including the second hash value;
 - storing the compound hash value in the trusted hardware device;
 - receiving a request for a VMM service that is associated with the first VM, wherein the request comprises a challenger hash value;
 - computing a current compound hash value based on a combination of the first VM platform configuration including the challenger hash value and the second VM platform configuration including the second VM hash value;
 - determining whether the current compound hash value is equal to the stored compound hash value; and
 - executing the received request when the current compound hash value is equal to the stored compound hash value.

2. (Original) The method of claim 1 wherein the VMM is loaded from firmware, the firmware including instructions compliant with an Extensible Firmware Interface (EFI) specification.
3. (Original) The method of claim 1 wherein sharing the trusted hardware device comprises multiplexing a first request from the first VM and a second request from the second VM to the trusted hardware device using the VMM multiplexer.
4. Canceled.
5. Canceled.
6. Canceled.
7. (Previously Presented) The method of claim 1, further comprising sealing secret information from the first VM with the compound platform configuration using the trusted hardware device.
8. (Original) The method of claim 7, further comprising unsealing the secret information using the trusted hardware device if a current first VM platform configuration matches the first VM platform configuration.
9. (Original) The method of claim 1, further comprising maintaining a queue of trusted hardware device requests by the VMM multiplexer, the trusted hardware requests sent to the trusted hardware device from the first VM and the second VM.

10. (Original) The method of claim 9, further comprising reporting a first request from the first VM is in progress when the trusted hardware device is polled by the first VM regarding the status of the first request, the first request actually waiting in the queue to be processed by the trusted hardware device.

11. (Original) The method of claim 1 wherein the trusted hardware device includes a trusted platform module (TPM).

12. (Previously Presented) An article of manufacture comprising:
a storage device including a plurality of instructions which when executed perform operations comprising:
loading a virtual machine monitor (VMM) in a computer system to support a first virtual machine (VM) and a second VM;
loading the first VM and the second VM;
receiving a first VM platform configuration including information measured from the first VM;
computing a first virtual hash based on the first VM platform configuration;
receiving a second VM platform configuration including information measured from the second VM;
computing a second virtual hash value based on the second VM platform configuration;
sending the first virtual hash value and the second virtual hash value to a trusted platform module (TPM) of the computer system, the TPM to compute a compound hash value based on the first virtual hash value and the second virtual hash value;
receiving a request for a VMM service that is associated with the first VM, wherein the request comprises a challenger hash value;

computing a current compound hash value based on a combination of the first VM platform configuration including the challenger hash value and the second VM platform configuration including the second VM hash value;

determining whether the current compound hash value is equal to the stored compound hash value; and

executing the received request when the current compound hash value is equal to the stored compound hash value.

13. Canceled.

14. Canceled.

15. (Previously Presented) The article of manufacture of claim 12 wherein execution of the plurality of instructions further perform operations comprising sending a seal command to the TPM to seal secret information from the first VM with the compound hash value.

16. (Original) The article of manufacture of claim 15 wherein execution of the plurality of instructions further perform operations comprising sending an unseal command to the TPM from the first VM to unseal secret information associated with the first VM.

17. (Original) The article of manufacture of claim 12 wherein execution of the plurality of instructions further perform operations comprising maintaining a TPM request queue to queue a first TPM request from the first VM and a second TPM request from the second VM.

18. (Original) The article of manufacture of claim 17 wherein execution of the plurality of instructions further perform operations comprising reporting the second TPM request is in progress if the TPM is polled by the second VM, the second TPM request actually waiting in the TPM request queue.

19. (Previously Presented) A computer system, comprising:

a processor;

a trusted hardware device operatively coupled to the processor; and

at least one flash memory device operatively coupled to the processor, the at least one flash memory device including firmware instructions which when executed by the processor perform operations comprising:

loading a virtual machine monitor (VMM) in the computer system to support a first virtual machine (VM) and a second VM, the VMM including a VMM multiplexer;

loading the first VM and the second VM;

maintaining a first VM platform configuration including a first hash number and a second VM platform configuration including a second hash number by the VMM multiplexer; and

storing a compound hash number corresponding to a platform configuration based on a combination of the first VM platform configuration and the second VM platform configuration in the trusted hardware device;

receiving a request for a VMM service that is associated with the first VM, wherein the request comprises a challenger hash value;

computing a current compound hash value based on a combination of the first VM platform configuration including the challenger hash value and the second VM platform configuration including the second VM hash value;

determining whether the current compound hash value is equal to the stored compound hash value; and

executing the received request when the current compound hash value is equal to the stored compound hash value.

20. Canceled.

21. (Original) The computer system of claim 19 wherein execution of the plurality of firmware instructions further perform operations comprising maintaining a queue of trusted hardware device requests by the VMM multiplexer, the trusted hardware device requests sent to the trusted hardware device from the first VM and the second VM.

22. (Original) The computer system of claim 19 wherein the firmware instructions compliant with an Extensible Firmware Interface (EFI) specification.

23. (Original) The computer system of claim 19 wherein the trusted hardware device comprises a trusted platform module (TPM).

24. (Currently amended) A method, comprising:
loading an untrusted virtual machine monitor (VMM) to support a plurality of virtual machines in a computer system, the VMM including a VMM multiplexer;
loading a first and a second virtual machine (VM) supported by the VMM;
sharing a trusted hardware device between the first VM and the second VM using the VMM multiplexer;
receiving a request for a VMM service that is associated with the first VM,
wherein the request comprises a challenger hash value;

determining a first VM platform configuration including a first hash value based on information measured from the first VM and a second VM platform configuration including a second hash value based on information measured from the second VM;

using a trusted hardware device shared between the first and the second VM to determine a stored compound hash value based on a combination of the first VM platform configuration and the second VM platform configuration;

computing a current compound hash value based on a combination of the first VM platform configuration including the challenger hash value and the second VM platform configuration including the second VM hash value;

determining whether the current compound hash value is equal to the stored compound hash value; and

executing the received request when the current compound hash value is equal to the stored compound hash value.

25. (Previously Presented) The method of claim 24 wherein the VMM is loaded from firmware, the firmware including instructions compliant with an Extensible Firmware Interface (EFI) specification.

26. (Previously Presented) The method of claim 1 wherein sharing the trusted hardware device comprises multiplexing a first request from the first VM and a second request from the second VM to the trusted hardware device using the VMM multiplexer.

27. (Currently amended) The method of claim 24 wherein further comprising:
~~determining a first VM platform configuration including a first hash value based on information measured from the first VM and a second VM platform configuration including a second hash value based on information measured from the second VM;~~

~~using a trusted hardware device shared between the first and the second VM to
determine a compound hash value based on a combination of the first VM platform
configuration and the second VM platform configuration; and
storing the compound hash value is stored in the trusted hardware device.~~